

1/16

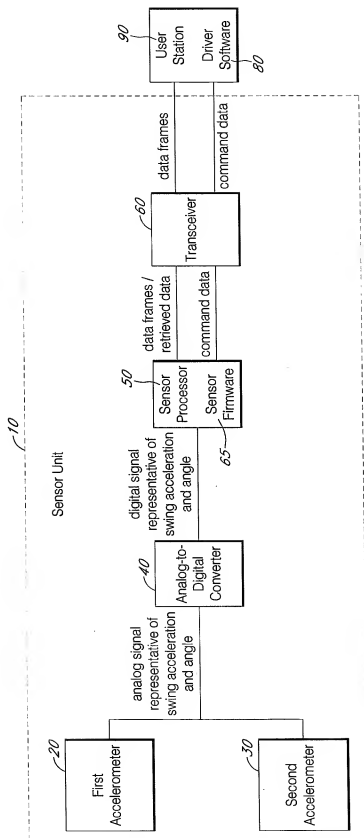


FIG. 1

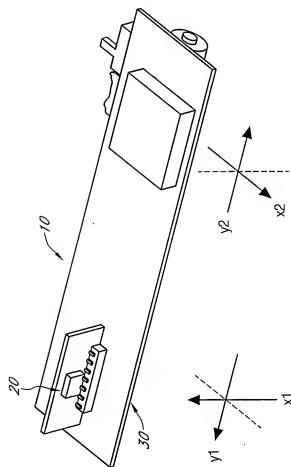


FIG. 2

3/16

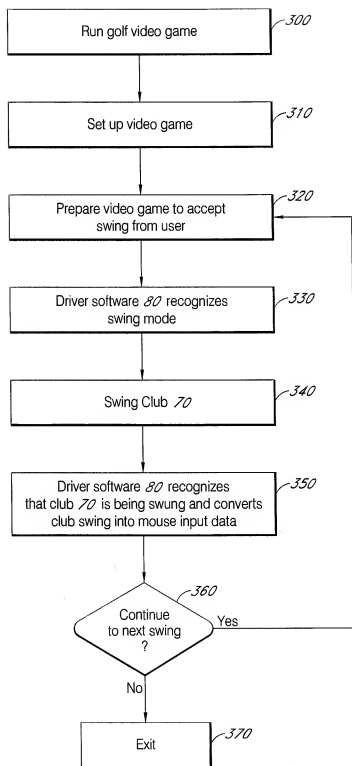


FIG. 3

4/16

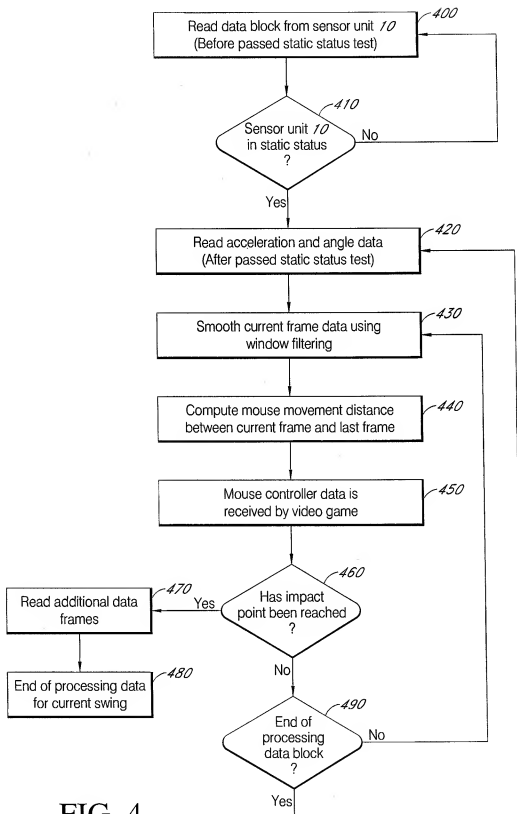


FIG. 4

5/16

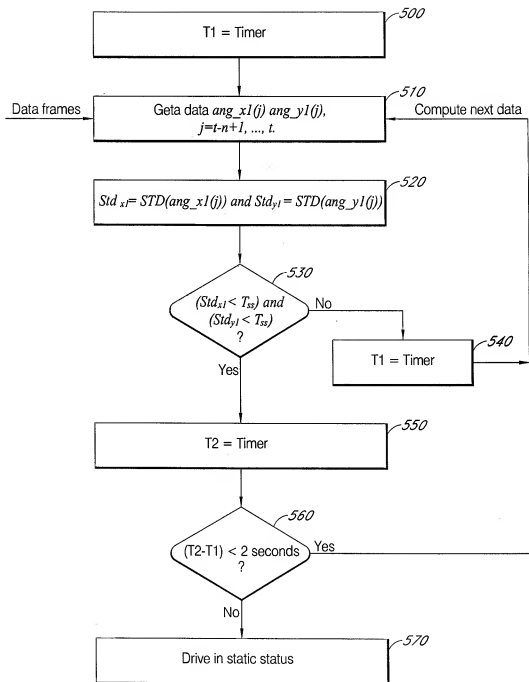


FIG. 5

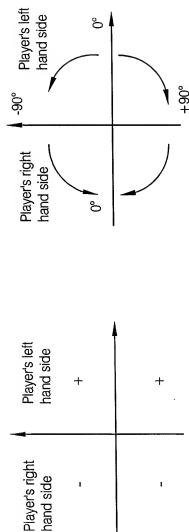


FIG. 6B

FIG. 6A

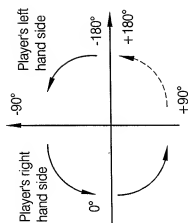


FIG. 6D

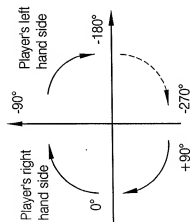


FIG. 6C

8/16

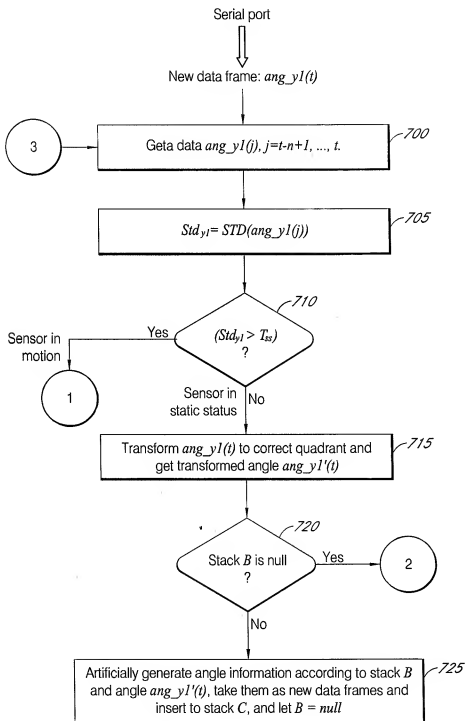


FIG. 7A

9/16

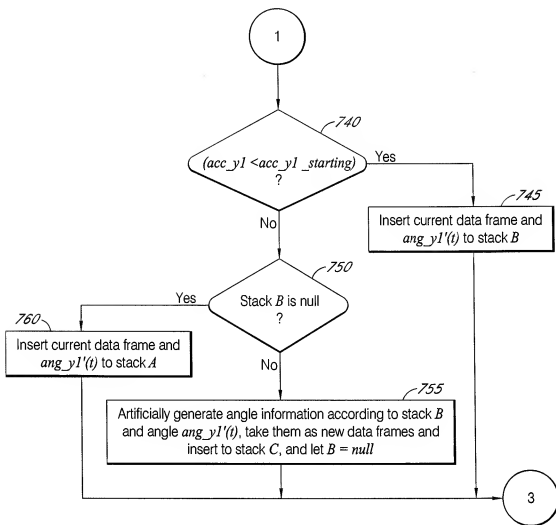


FIG. 7B

10/16

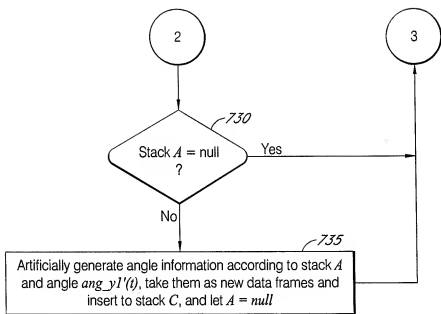


FIG. 7C

```
If ( ang_y1(t) >= 0 and club is swing up ) Then
  If ( ang_x2(t) > 0 ) Then ang_y1'(t) = - 180 - ang_y1(t)
  ElseIf ( ang_y1(t) > ang_y1_starting - 60 and club is swing down and ang_x1(t)
    <= 0 ) Then
    ang_y1'(t) = 180 - ang_y1(t)
  ElseIf ( ang_y1(t) <= 0 and club is swing up ) Then
    If ( ang_x2(t) >= 0 ) Then
      ang_y1'(t) = - 180 - ang_y1(t)
    ElseIf ( ang_x2(t) < 0 ) Then
      ang_y1'(t) = ang_y1(t)
    End If
  End If
End If
```

FIG. 8

12/16

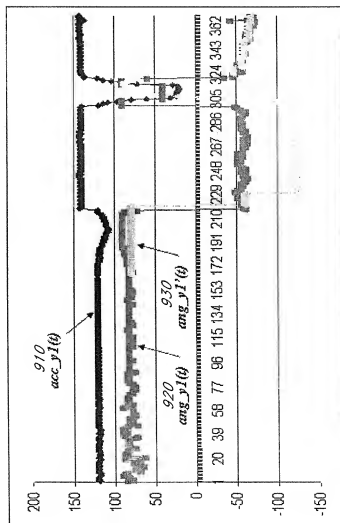


FIG. 9

```

Input: angle, change and current_angle; Output: distance
1) Let distance = angle_change
2) If (swing down And current_angle > 90) Then distance = distance * 2
3) If (swing up) Then
4) Suppose a) current_angle < 45 And current_angle >= -45; or b)
   current_angle < -45 And current_angle >= -90; or c) current_angle < -90 And
   current_angle >= -145; or d) current_angle < -145 And current_angle >= -
   180; or e) current_angle < -180. Then Let R = 1.25, 1.5, 5, 7, 10
   corresponding to a)-e) respectively.
5) Let distance = distance * R
6) End If
7) If (swing down) Then
8) Suppose a) current_angle <= -180; or b) current_angle <= -135 And
   current_angle < -180; or c) current_angle <= -90 And current_angle > -135; or
   d) current_angle >= -90 And current_angle <= 0; or e) current_angle > 0 And
   current_angle <= 30; or f) current_angle > 30 And current_angle <= 90. Then
   Let R = 12, 10, 8, 6, 5, 5 corresponding to a)-f) respectively.
9) Let distance = distance / R
10) adjust distance value according to acceleration acc_y1.
11) If (distance value is small) Then adjust it according to the club's position
12) End If
13) If (club is not in motion) Then Let distance = 0
14) If (club passed starting position And distance < 5) Then Let distance = 5
  
```

FIG. 10A

Input: angle_change and current_angle; Output: distance

```

1) distance = angle_change
2) Suppose a) current_angle > starting_angle - 15; or b) current_angle >
   starting_angle - 30 And current_angle <= starting_angle - 15; or c)
   current_angle > starting_angle - 45 And current_angle <= starting_angle - 30;
   or d) current_angle > starting_angle - 60 And current_angle <= starting_angle
   - 45; or e) otherwise. Then Let R = 12, 12, 8, 8, 4 corresponding to a)-e) respectively.
3) Let distance = distance * R
4) If (swing down) Then
5)   adjust distance value according to acceleration acc_y1.
6) If (distance value is small) Then adjust it according to the club's position
7) End If
8) If (club is not in motion) Then Let distance = 0
9) If (club passed starting position And distance < 5) Then Let distance = 5
  
```

FIG. 10B

Input: angle_change and current_angle; Output: distance

```

1) distance = angle_change
2) Suppose a) current_angle > starting_angle - 15; or b) current_angle >
   starting_angle - 30 And current_angle <= starting_angle - 15; or c)
   current_angle > starting_angle - 45 And current_angle <= starting_angle - 30;
   or d) current_angle > starting_angle - 60 And current_angle <= starting_angle
   - 45; or e) otherwise. Then Let R = 24, 24, 16, 16, 8 corresponding to a)-e)
   respectively.
3) Let distance = distance * R
4) If (swing down) Then
5)   adjust distance value according to acceleration acc_y1.
6)   If (distance value is small) Then adjust it according to the club's position
7)   End If
8)   If (club is not in motion) Then Let distance = 0
9)   If (club passed starting position And distance < 5) Then Let distance = 5

```

FIG. 10C

Input: distance, loop0 and distance_number
Suppose club is in a) Putting status; or b) Chipping status; or c) Full swing status. **Then** Let R = MAX_LOOP_STEP_PUTT;
MAX_LOOP_STEP_CHIP, MAX_LOOP_STEP_NORMAL, respectively.

- 1) **For** k = 0 **To** distance_number-1
- 2) distance_number = distance / R
- 3) **For** k = 0 **To** distance_number-1
- 4) distance_loop(k) = R
- 5) **Next** k
- 6) **If** (distance_number >= 1) **Then**
- 7) distance_number = distance_number - 1
- 8) **Else**
- 9) distance_loop(distance_number) = distance
- 10) **End If**

FIG. 11